Security Principles: Defenses

SPR500 NBB

Project

Vulnerability Assessment Report

Group 2	
Student Name	**Student ID**
Khondoker Ishmum Muhammad	155895212
Yash Siraj Devani	160409215
Ananthu Krishna Vadakkeppatt	154290217
Aryan Santosh Saindane	136235215

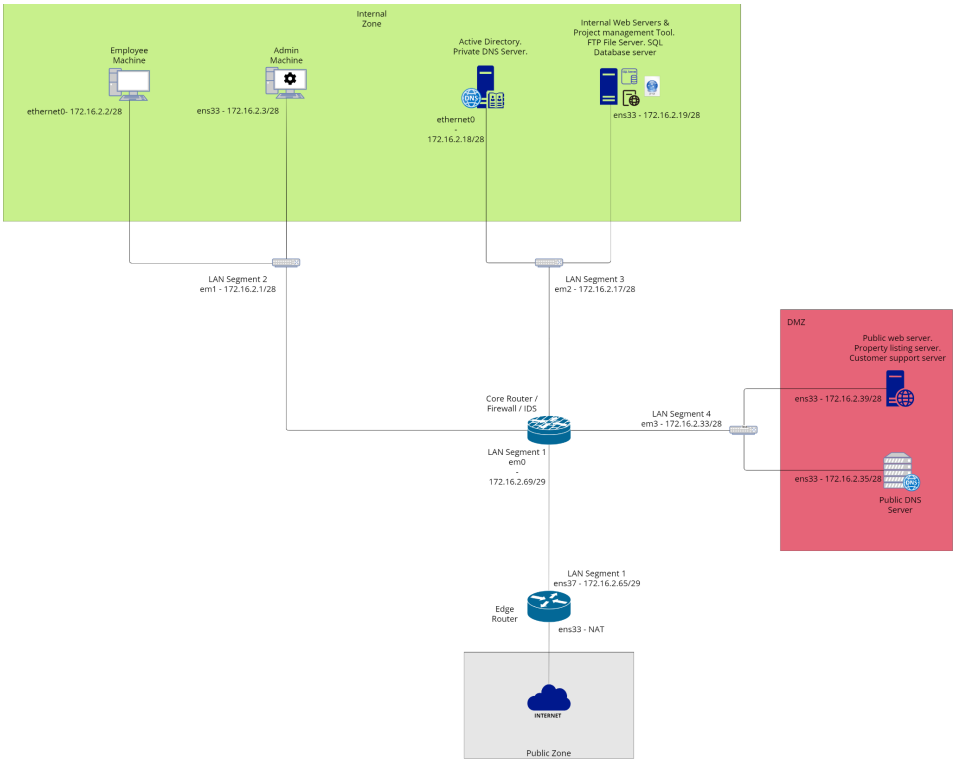
Table Of Contents

Executive Summary

This report provides a comprehensive overview of vulnerabilities identified within Agricore's commerce infrastructure through a thorough vulnerability assessment. The assessment, conducted using Nessus and NMAP scanning tools, revealed key insights into the security posture of Agricore's network. The report categorizes and documents vulnerabilities based on severity levels, offering a clear understanding of potential risks. Mitigation strategies and recommendations are discussed to address and remediate these vulnerabilities effectively.

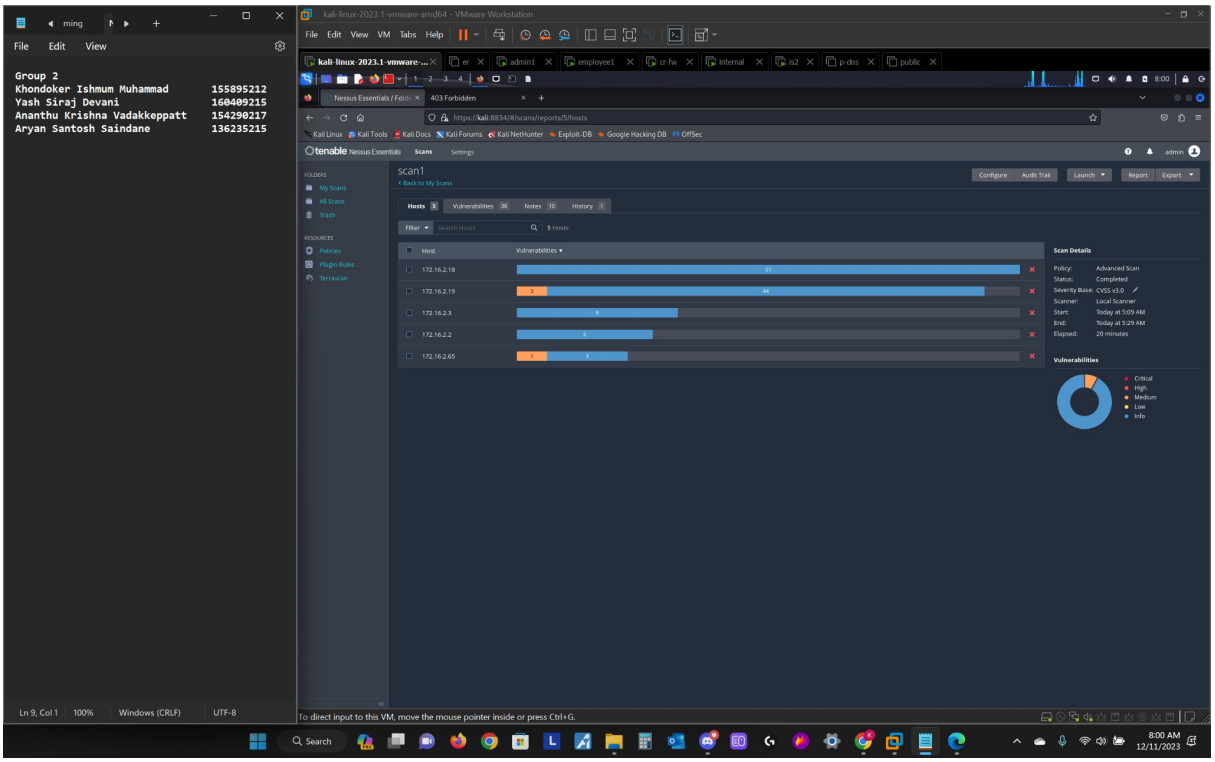
Scope

The main purpose of the assessment is to examine the security measures implemented on the website www.xeedie.com In order to enhance the protection of business assets a comprehensive approach was taken, utilizing scanning tools, like Nessus. These tools were strategically used to analyze the web applications, databases and the complex network structure along with its components This evaluation covers a range, including both external systems aiming to have a holistic understanding of the security posture. The goal is to identify vulnerabilities that could jeopardize the integrity of business assets. To achieve this goal a combination of vulnerability scanning tools and analysis of system configurations was carried out. The selected tools were specifically chosen for their effectiveness in detecting a range of security issues. By leveraging this set of scanning technologies and configuration analysis we aim to uncover any weaknesses and mitigate threats thus strengthening the resilience and security of the evaluated systems.

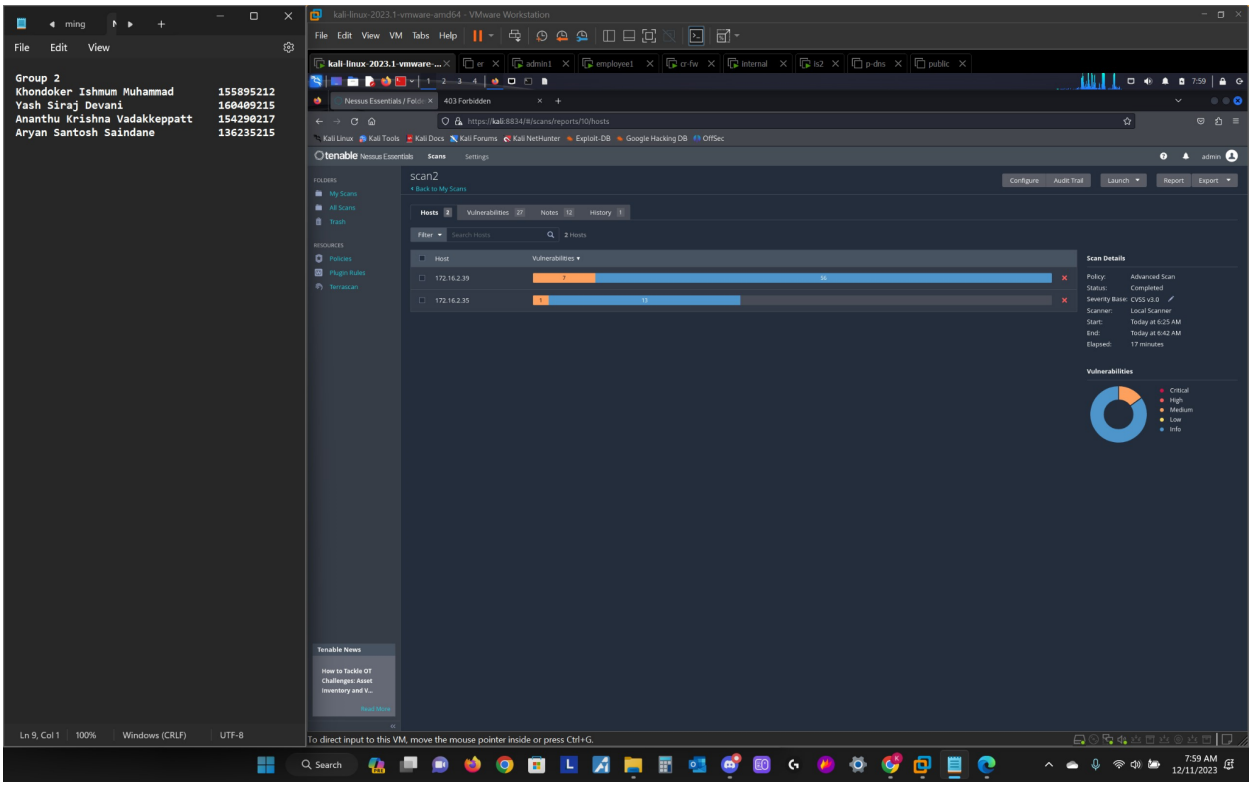


Nessus Scans With Web App Scanning Enabled

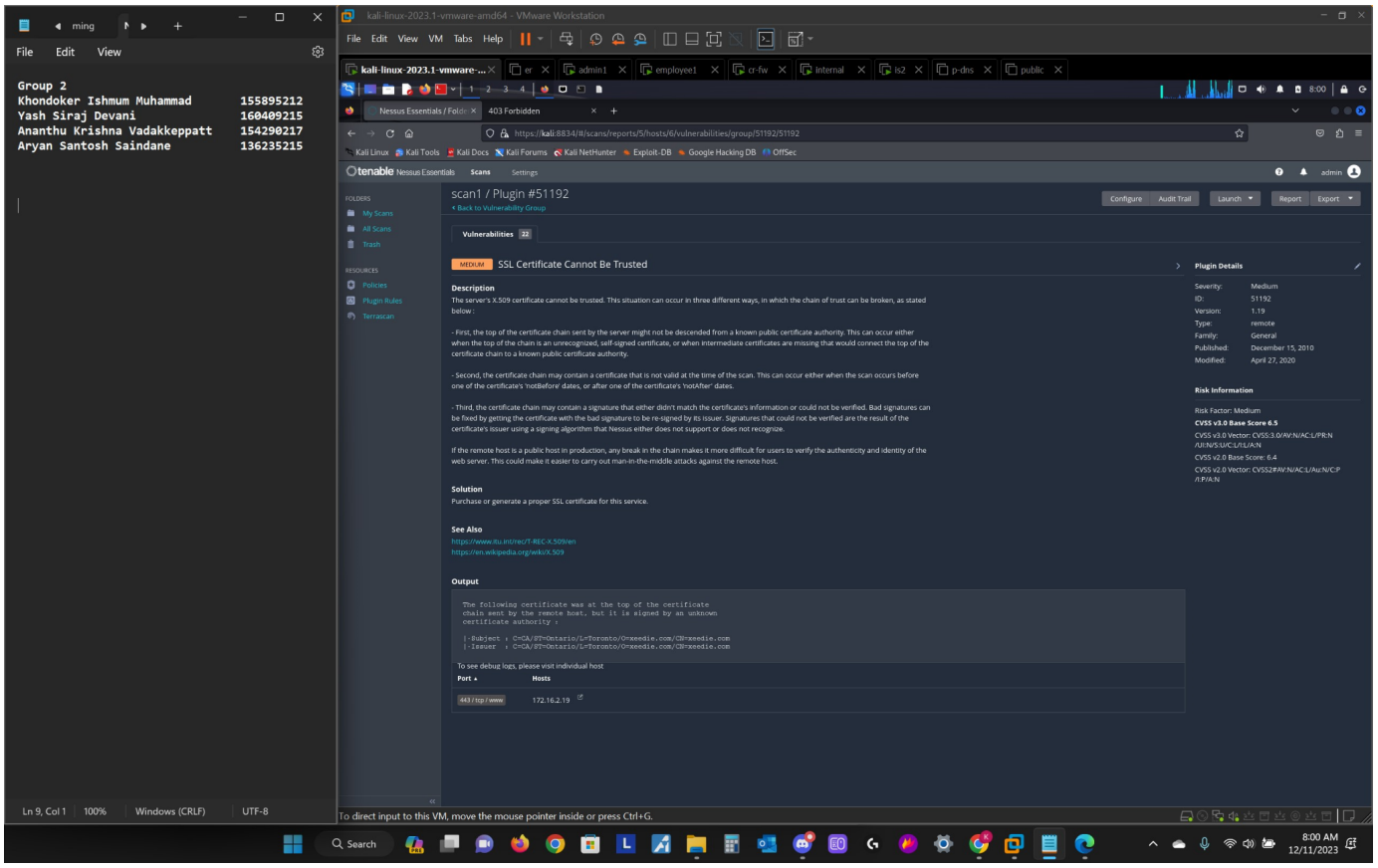
Below we have our nessus scans for the internal machines.



Below we have the nessus scans for the machines in the DMZ and are public facing.



SSL Certificate Cannot Be Trusted



Description:

The X.509 certificate issued by the server cannot be trusted. This circumstance can arise in three ways, as detailed below, in which the chain of trust can be broken:

- First, the server's certificate chain may not be descended from a known public certificate authority. This can happen when the top of the certificate chain is an unrecognized, self-signed certificate, or when intermediate certificates connecting the top of the certificate chain to a known public certificate authority are absent.

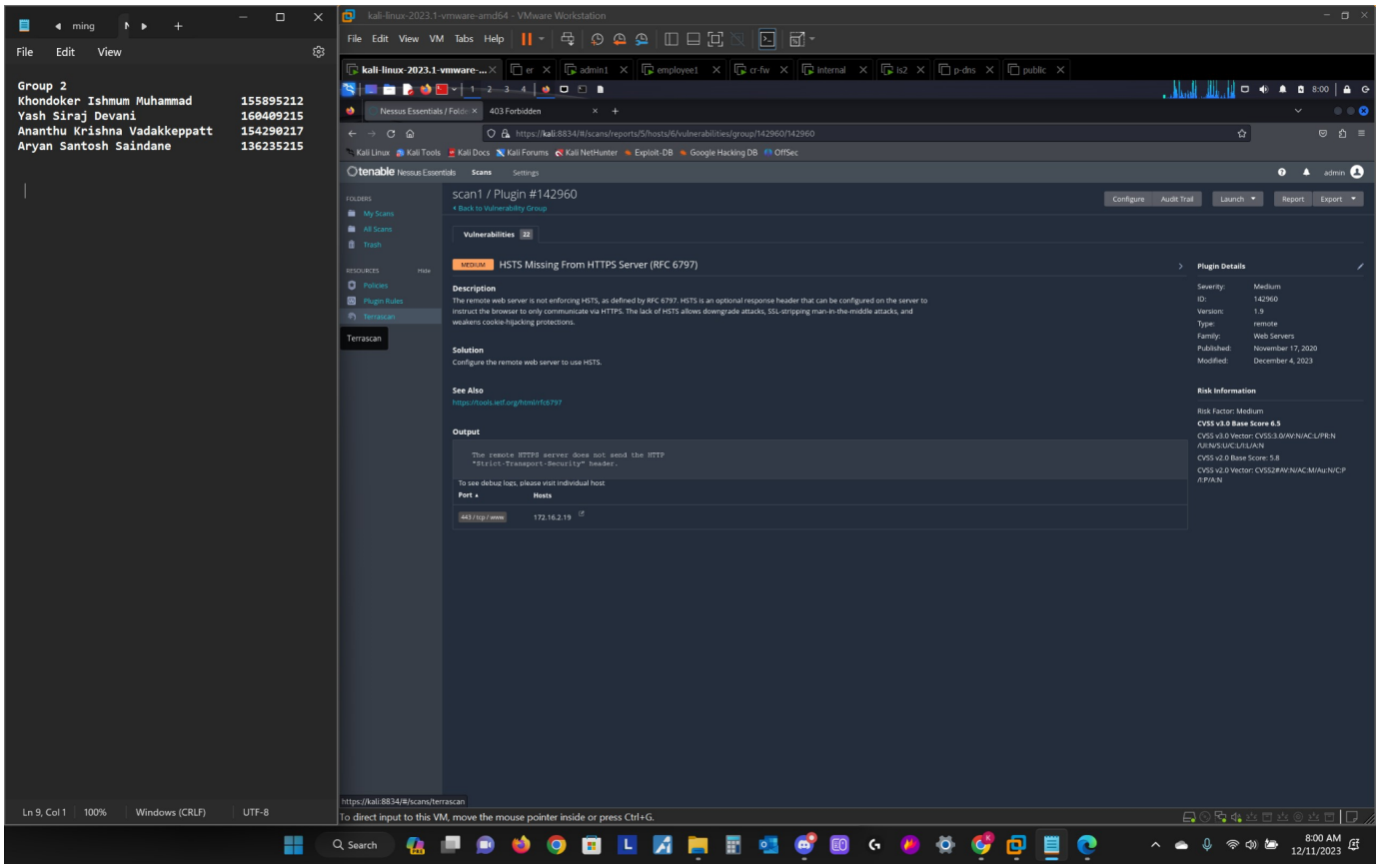
- Second, a certificate in the certificate chain may be invalid at the time of the scan. This can happen if the scan happens before or after one of the certificate's 'notBefore' or 'notAfter' dates.

- Third, the certificate chain may contain a signature that does not match the information in the certificate or cannot be verified. Bad signatures can be repaired by having the certificate with the bad signature re-signed by the certificate's issuer. Signatures that could not be verified are the result of the certificate's issuer using a signing algorithm that Nessus either does not support or does not recognize.

Severity: Medium

Scope: 172.16.2.19

HSTS Missing From HTTPS Server (RFC 6797)



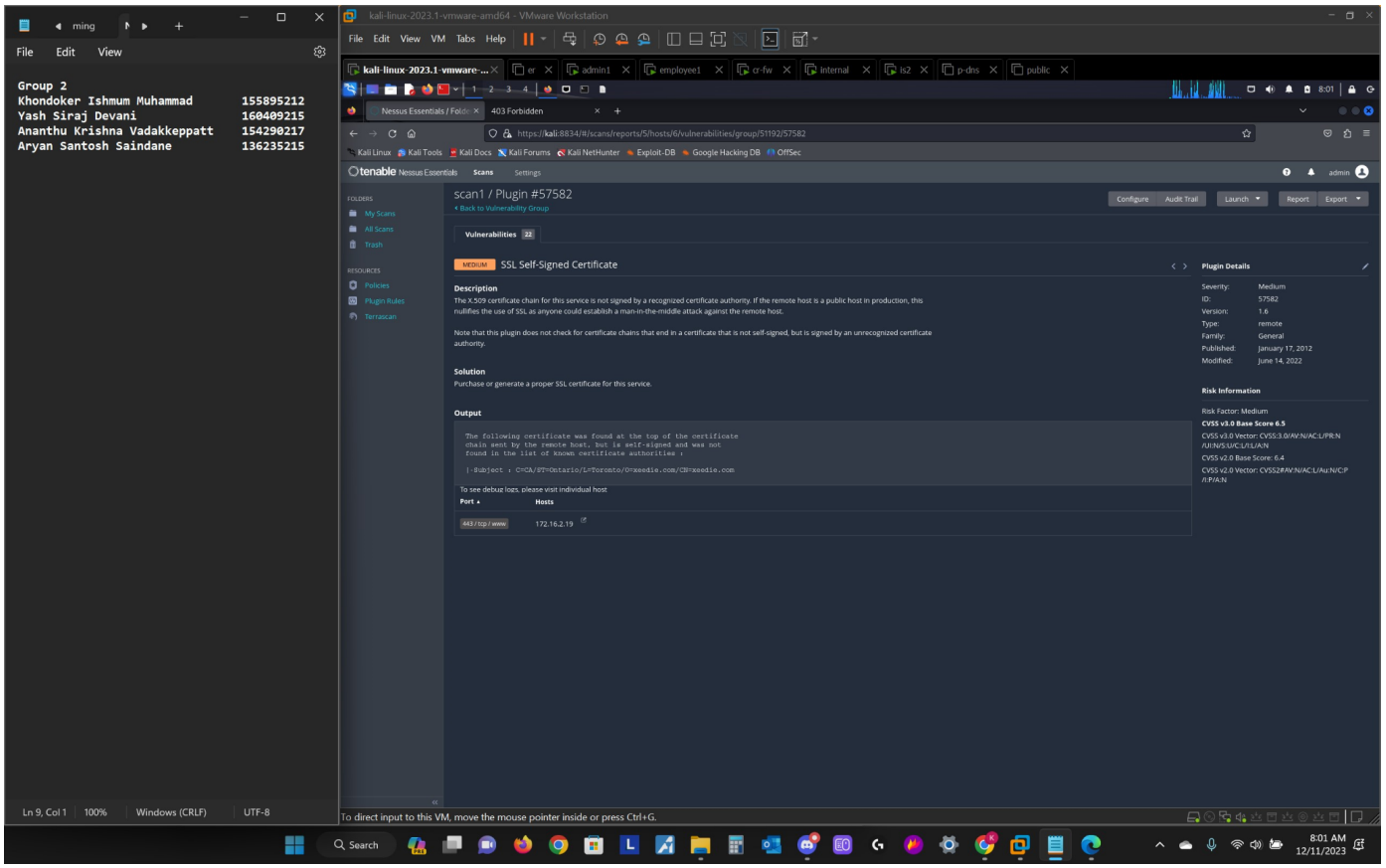
Description:

The remote web server is not enforcing HSTS in accordance with RFC 6797. HSTS is a response header that can be set on the server to advise the browser to only connect via HTTPS. The absence of HSTS allows downgrade attacks, SSL-stripping man-in-the-middle attacks, and cookie-hijacking defenses to be compromised.

Severity: Medium

Scope: 172.16.2.19

SSL Self Signed Certificate



Description:

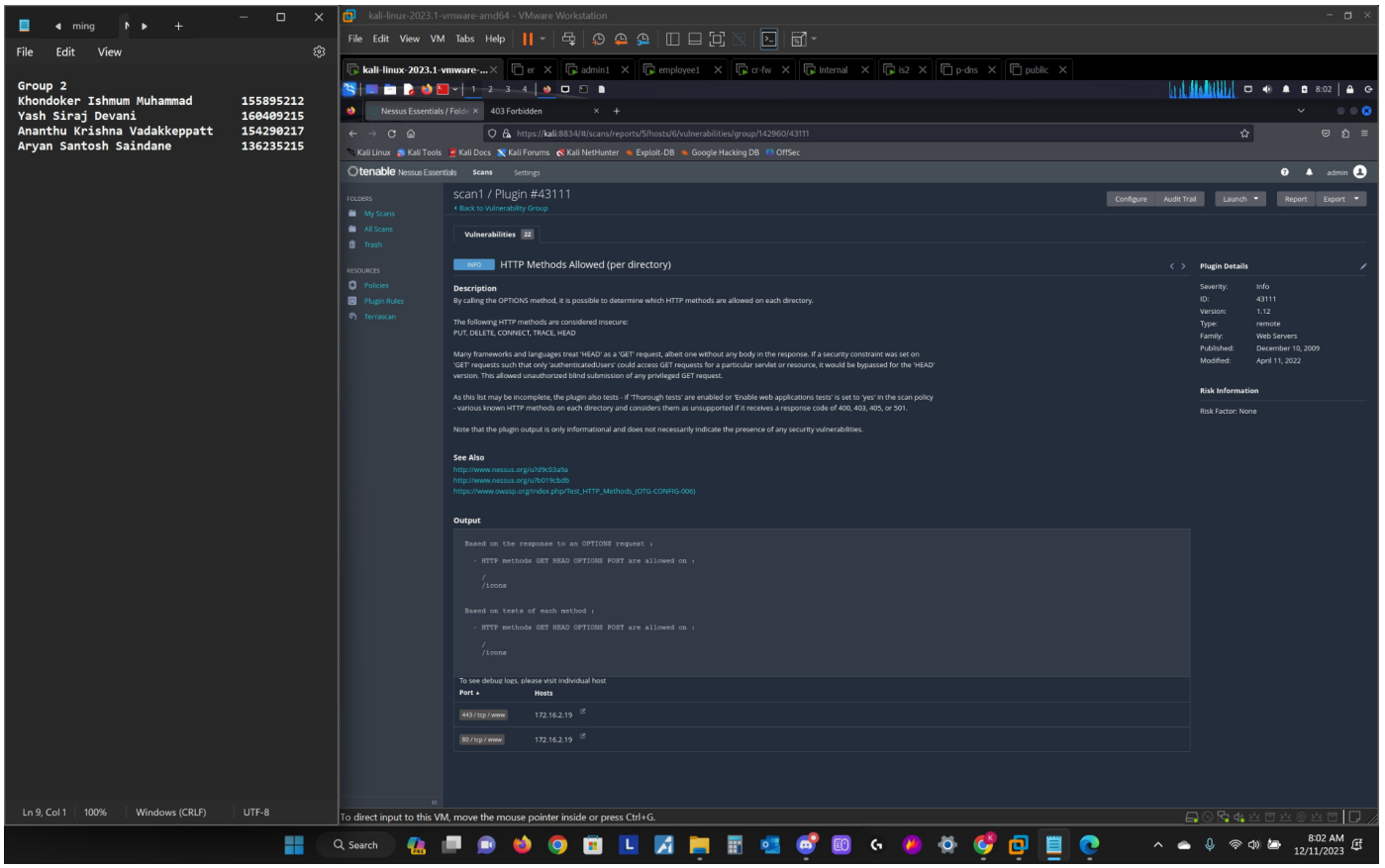
This service's X.509 certificate chain is not signed by a recognised certificate authority. If the remote host is a public host in production, SSL is rendered useless because anyone can launch a man-in-the-middle attack against it.

It should be noted that this plugin does not look for certificate chains that end in a certificate that is not self-signed but is signed by an unknown certificate authority.

Severity: Medium

Scope: 172.16.2.19

HTTP Methods Allowed (per directory)



Description:

It is possible to determine which HTTP methods are allowed on each directory by calling the OPTIONS method.

The following HTTP methods are deemed unsafe:
PUT, DELETE, CONNECT, TRACE, and HEAD are all commands.

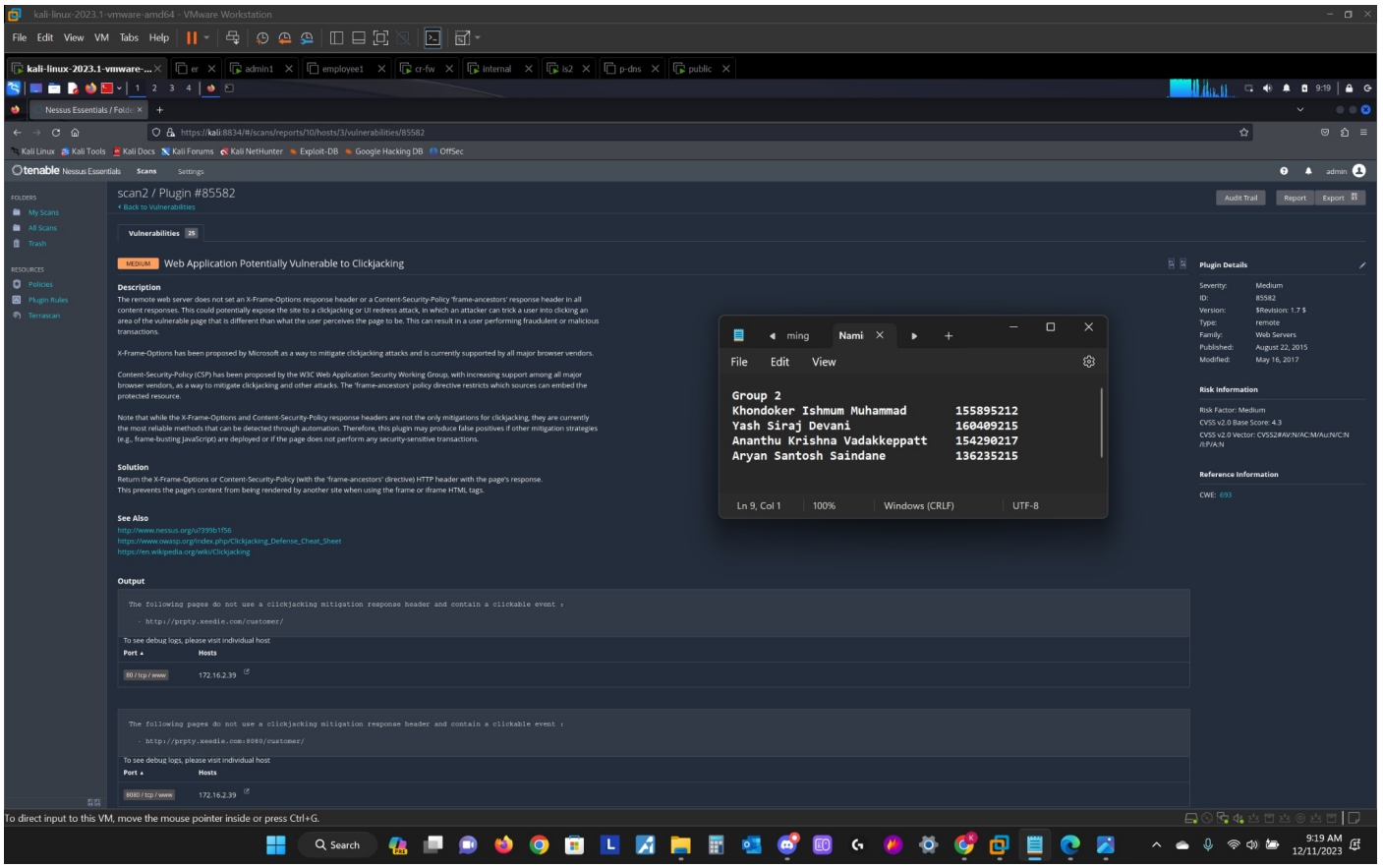
Many frameworks and languages consider 'HEAD' to be a 'GET' request, albeit one with no body in the response. If a security constraint on 'GET' requests was set so that only 'authenticatedUsers' could access GET requests for a specific servlet or resource, it would be ignored for the 'HEAD' version. This enabled the unauthorized submission of any privileged GET request blindly.

Because this list may be incomplete, if 'Thorough tests' are enabled or 'Enable web applications tests' is set to 'yes' in the scan policy, the plugin also tests various known HTTP methods on each directory and considers them unsupported if it receives a response code of 400, 403, 405, or 501.

Severity: Info

Scope: 172.16.2.19

Web Application Potentially Vulnerable To Clickjacking



Description:

In all content responses, the remote web server does not set an X-Frame-Options response header or a Content-Security-Policy 'frame-ancestors' response header. This could expose the site to a clickjacking or URL redress attack, in which an attacker can trick a user into clicking on a different area of the vulnerable page than the user perceives the page to be. As a result, a user may engage in fraudulent or malicious transactions.

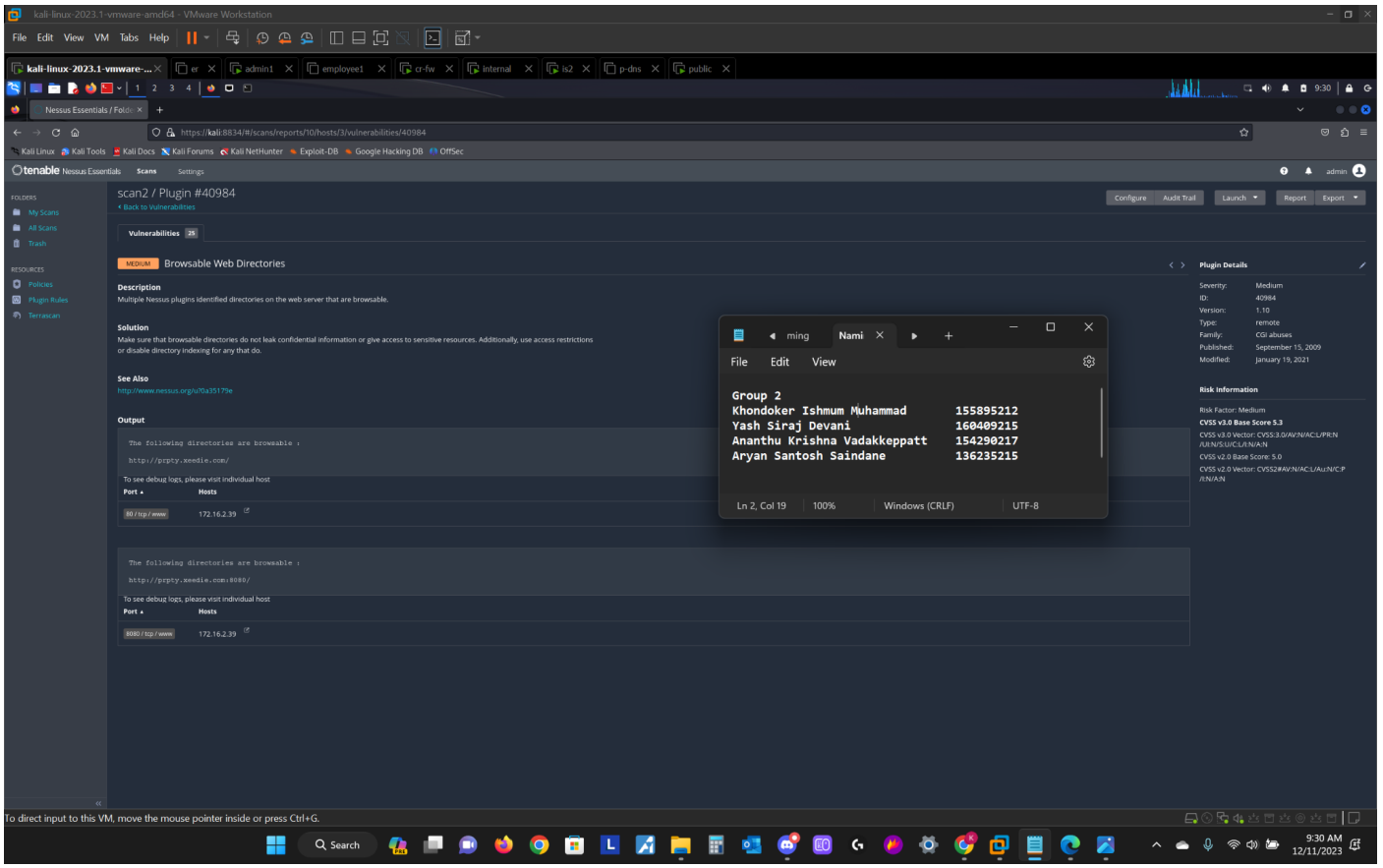
Microsoft proposed X-Frame-Options as a way to mitigate clickjacking attacks, and it is now supported by all major browser vendors.

The W3C Web Application Security Working Group proposed Content-Security-Policy (CSP) as a way to mitigate clickjacking and other attacks, and it is gaining support from all major browser vendors. The 'frame-ancestors' policy directive limits the sources from which the protected resource can be embedded.

Severity: Medium

Scope: 172.16.2.39

Browsable Web Directories



Description:

This can pose a security risk because sensitive information may be unintentionally exposed. Allowing browsing of directories that contain files that should not be publicly accessible, such as configuration files, databases, or personal information, could potentially lead to unauthorized access or data leakage.

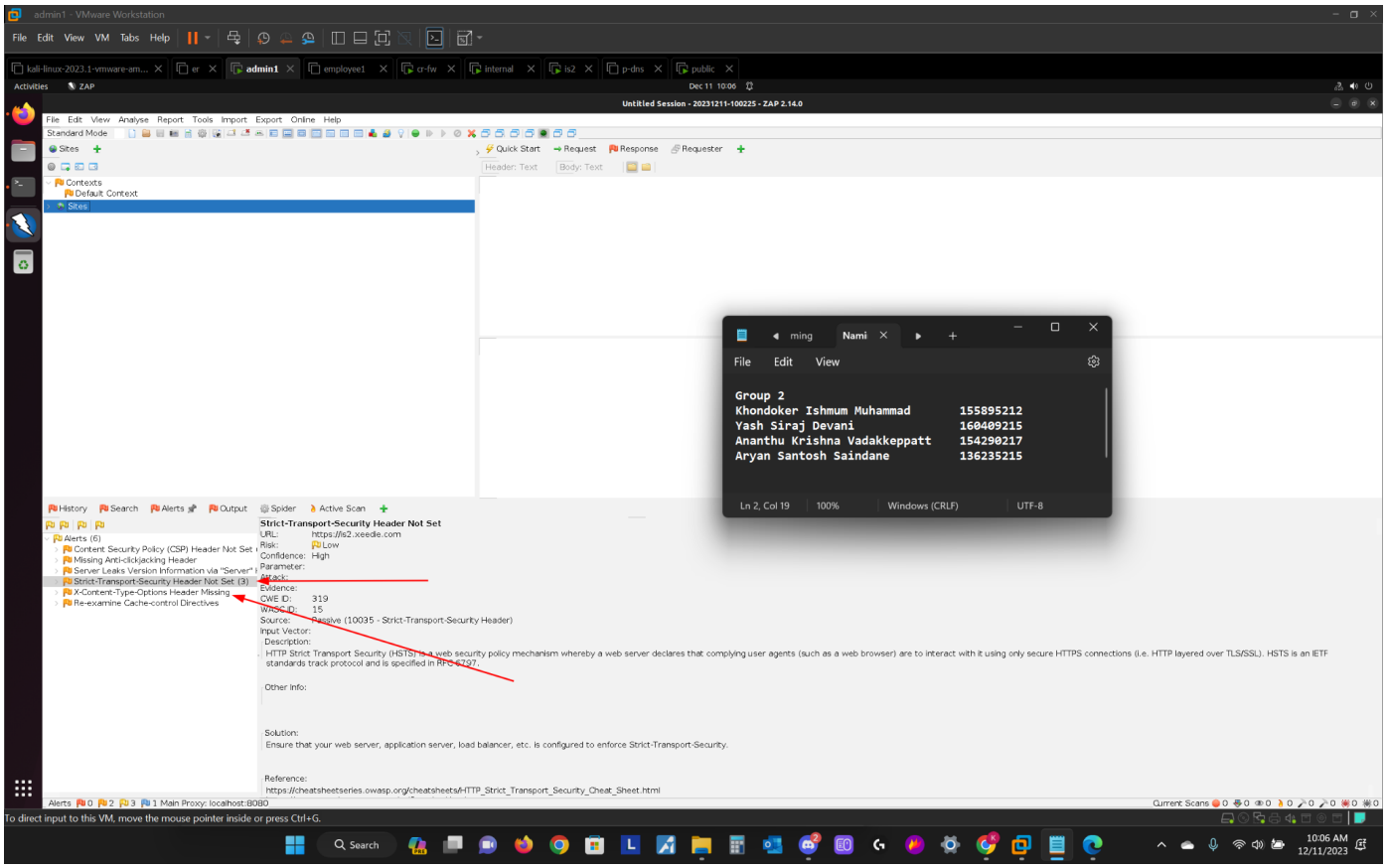
The Nessus scan identifies this vulnerability to alert system administrators or website owners to take action, such as adjusting server configurations or permissions, to prevent unauthorized access to these directories and thus improve web server security overall.

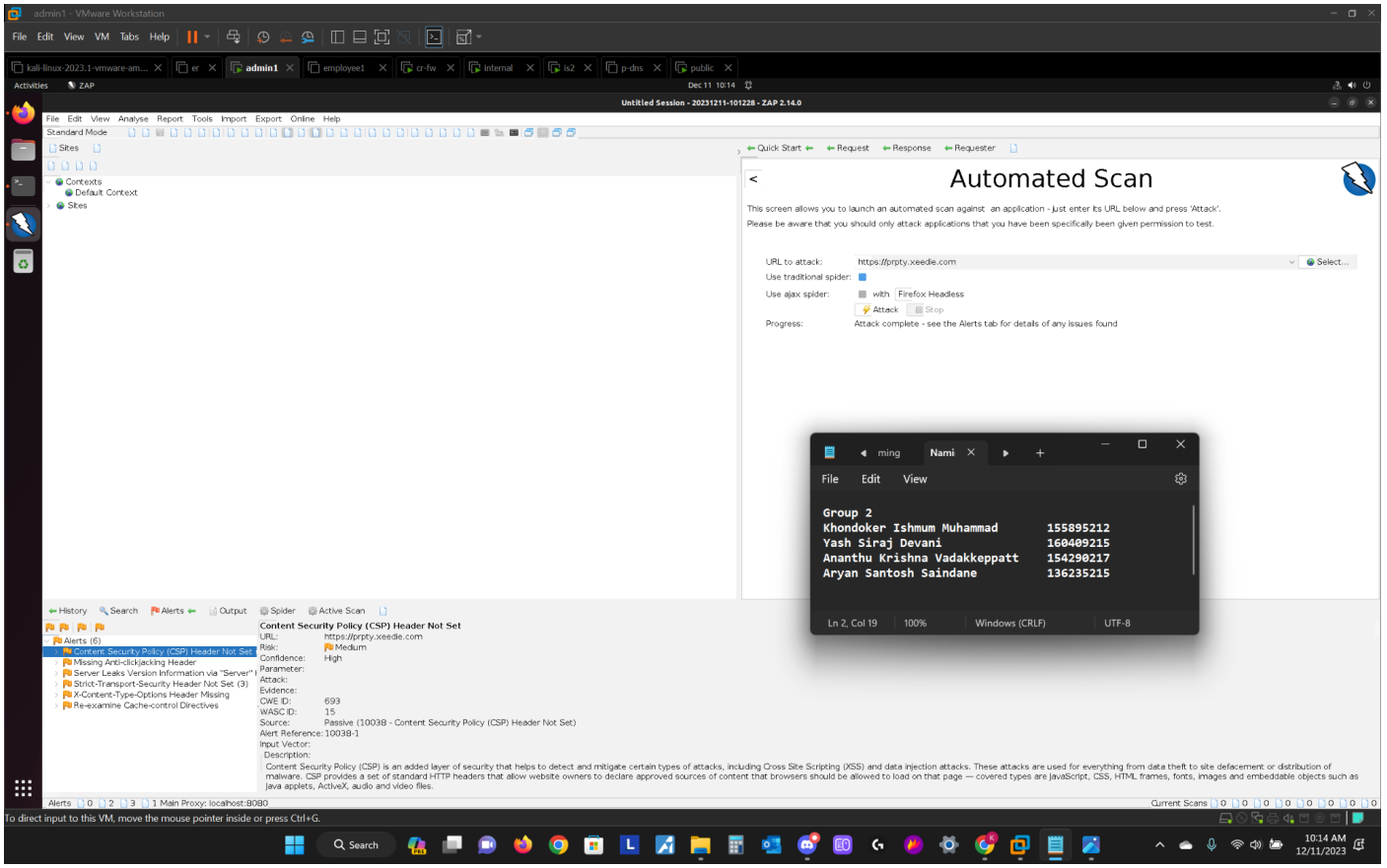
Severity: Medium

Scope: 172.16.2.39

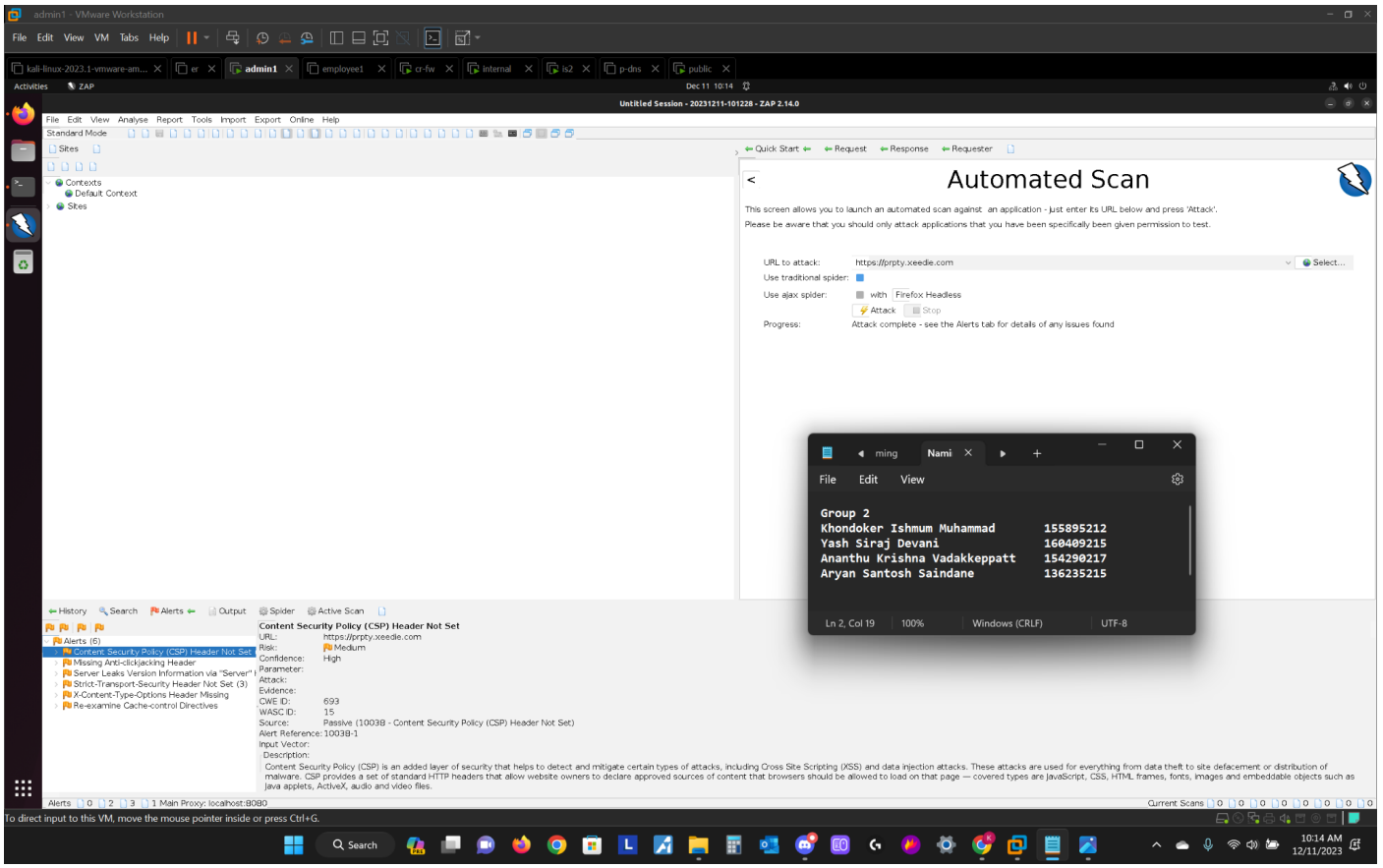
Zapp Scans

Below we see zap scans for further web app scanning.





Content Security Policy CSP Header Not Set

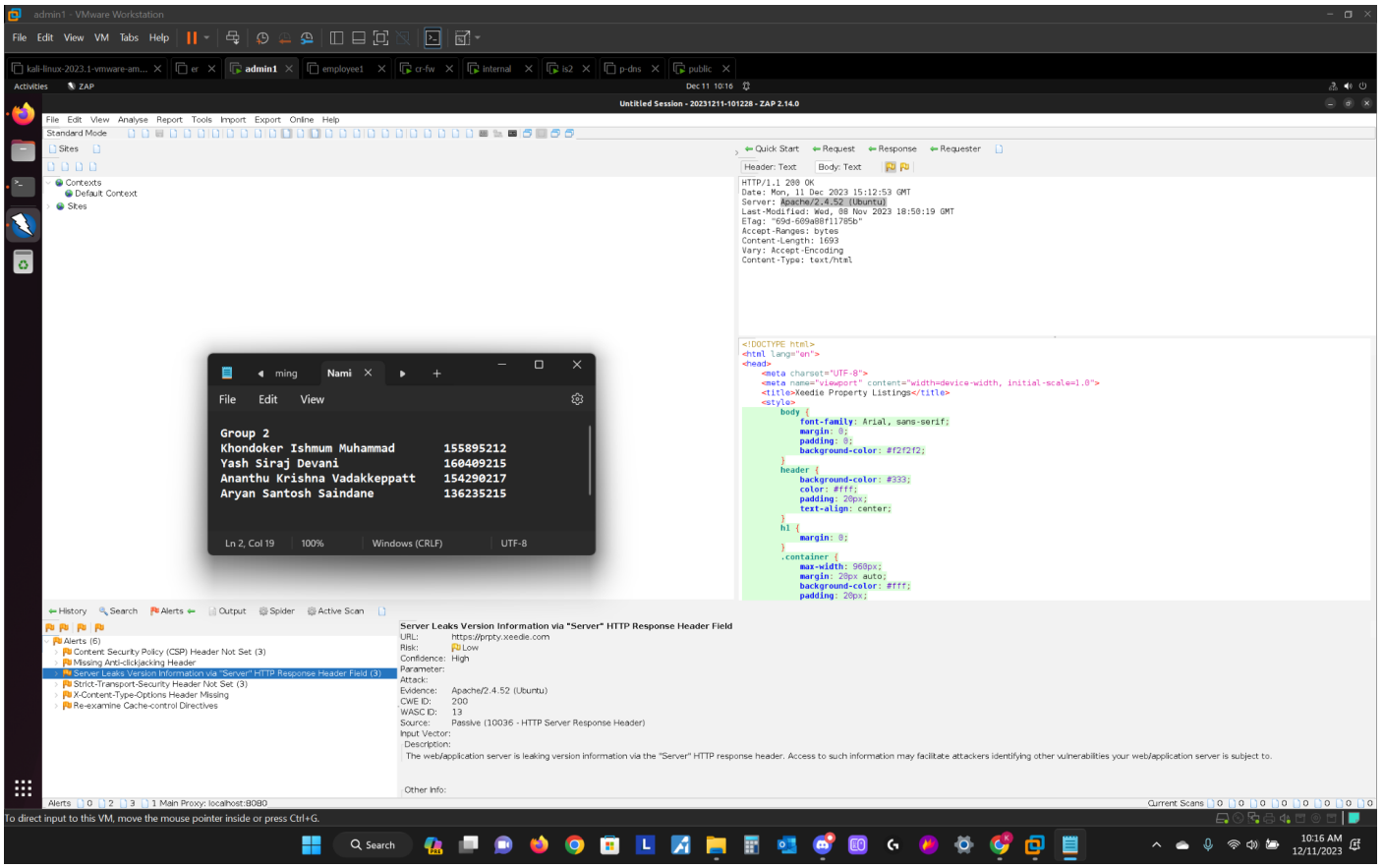


CSP Is another layer of security which can be used to help detect and mitigate attacks of several types that also includes cross-site scripting and data injection attacks. in this case it looks like the content security policy or CSP for short header is not set.

Scope: 172.16.2.19, 172.16.2.39

Risk/Severity: Medium

Server Leaks Information Via The Response Header

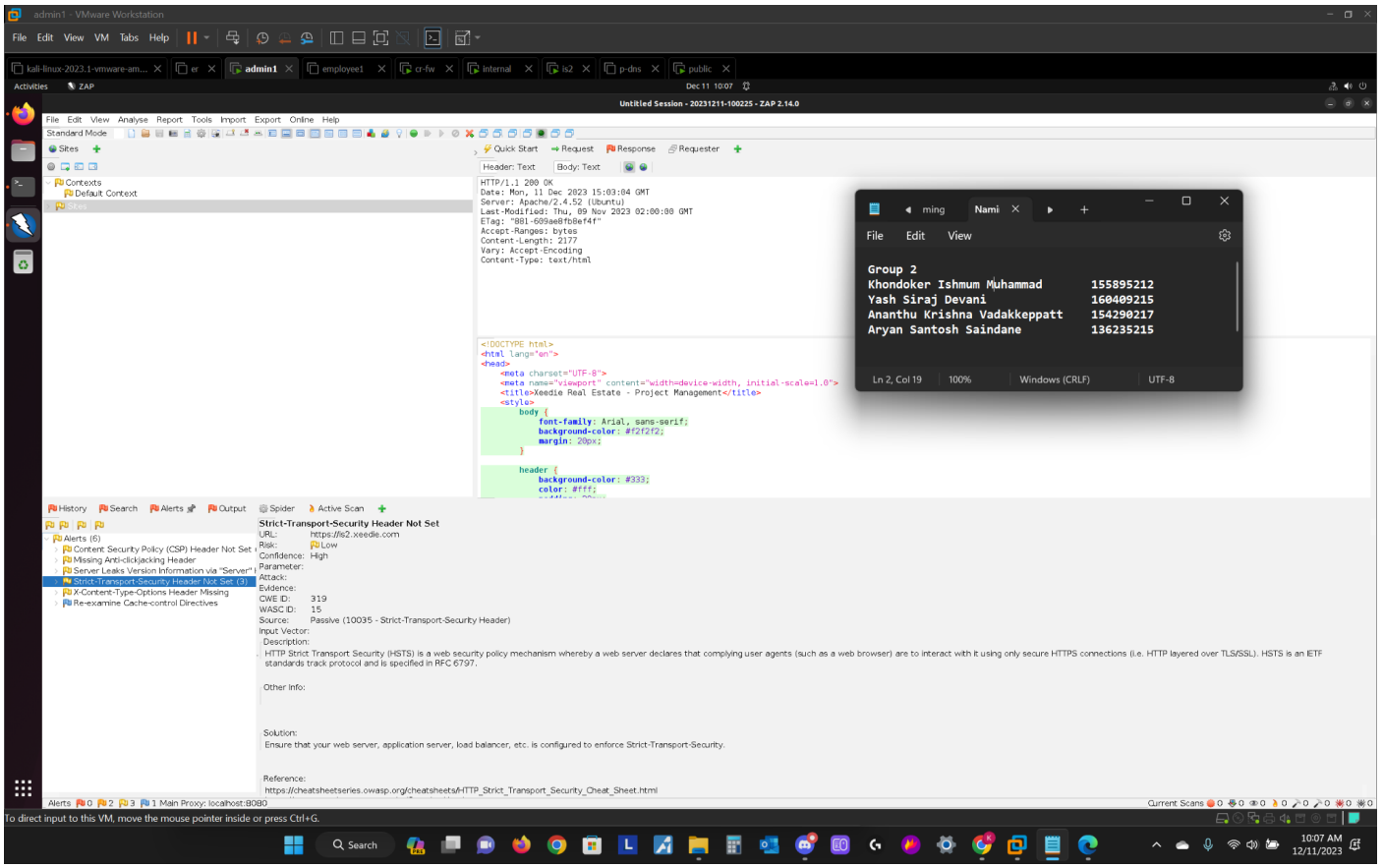


The web server is leaking information such as version information through the HTTP Response header of the server. attackers can use this information to find out ways to perform malicious attacks against the server and create plans of attacks against the server.

Scope: 172.16.2.19, 172.16.2.39

Risk/Severity: Low

Strict Transport Security Header Not Set (3)

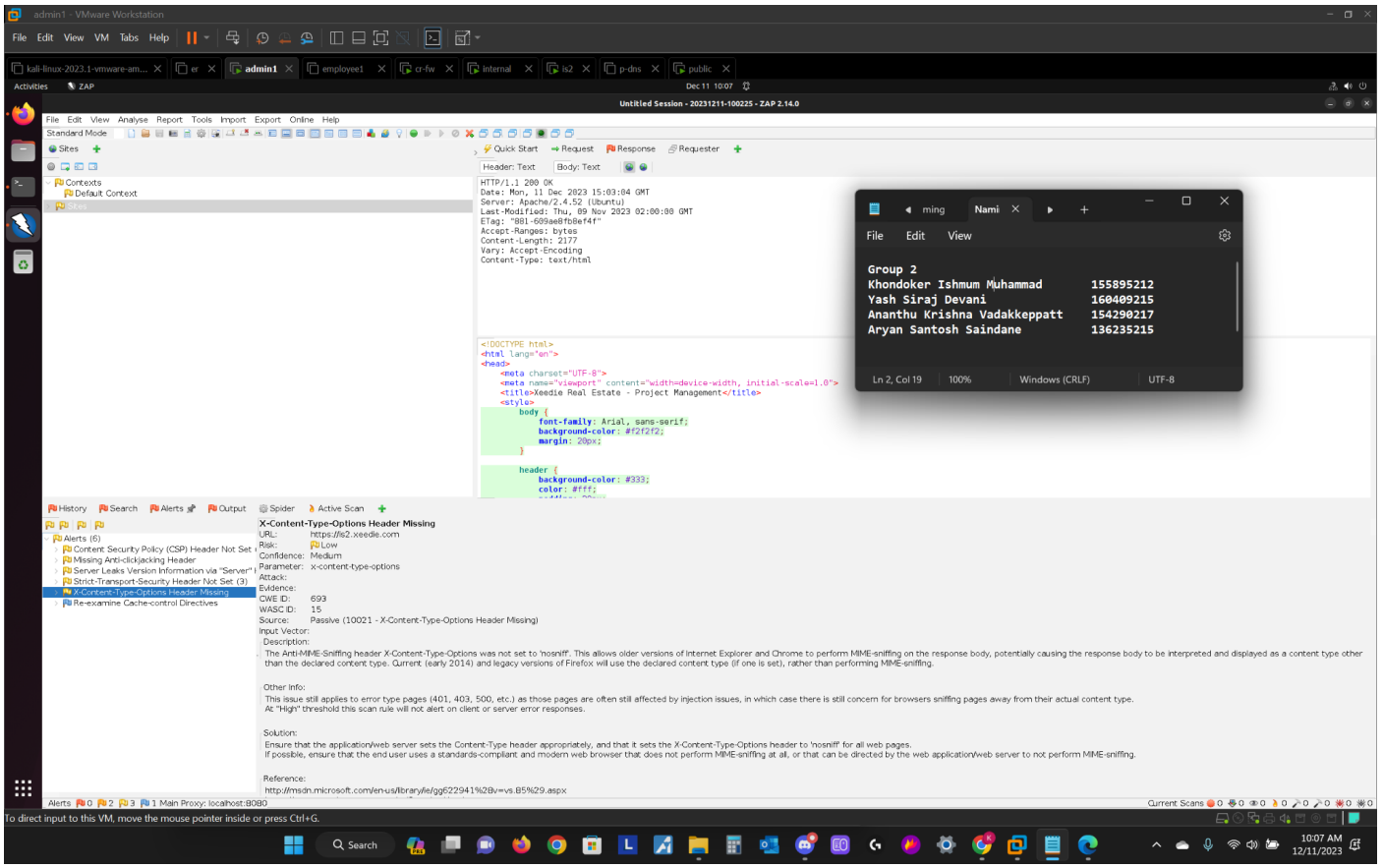


The HTTP strict Transport Security is a policy of web security where the mechanism is that a web server will declare that complying user agents such as a web browser should interact with https connections only. In this case, the strict Transport Security header is not set.

Scope: 172.16.2.19, 172.16.2.39

Risk/Severity: Low

X-Content Type Options Header Missing



The anti-miami sniffing header X content type options was not set to the option nosniff. this could allow the previous versions of browsers of Internet Explorer and chrome to perform in Miami sniffing on the response body. This could potentially cause the response body to the interpreted and displayed as a content type other than the declared content type.

Scope: 172.16.2.19, 172.16.2.39

Risk/Severity: Low

Mitigation Strategies

SSL Certificate Cannot Be Trusted: To make sure that our ssl certificates can be trusted the mitigation strategy is very simple. Just add the certificate and the root CA to the list of trusted certificates of Active Directory IIS services or import the certificate and the CA to the browser to ensure trust in the browser.

HSTS Missing From HTTPS Server (RFC 6797): To mitigate this issue, we just have to enable the HSTS in the configurations of the Web Server.So in the apache configurations we can add Header always set Strict-Transport-Security "max-age=63072000; includeSubDomains; preload" to your site's configuration.

SSL Self Signed Certificate: To make sure that our ssl certificates can be trusted the mitigation strategy is very simple. Just add the certificate and the root CA to the list of trusted certificates of Active Directory IIS services or import the certificate and the CA to the browser to ensure trust in the browser.

HTTP Methods Allowed (per directory): To mitigate this issue we need to find the http methods that are needed for the business processes and then make sure to restrict the methods that can be allowed.

Web Application Potentially Vulnerable To Clickjacking: To mitigate this issue, we can set up the X-Frame-Options header in your web server's response.

Browsable Web Directories: To mitigate this vulnerability, we can disable Directory Browsing in the configurations of the Web Server Configuration by using the options -Indexes in our .htaccess file or Apache configuration file to disable directory listing.

Content Security Policy CSP Header Not Set: With apache web server configurations we should implement content security policy. In the configurations we need to specify that we need to allow the browser to load, e.g., script-src, style-src.

Server Leaks Information via the Response Header: Setting ServerSignature Off and ServerTokens Prod will reduce the amount of data that is leaked through the server. These settings can be set in the apache configuration file or httpd.conf.

X-Content-Type-Options Header Missing: Here, we have to add the following configurations into our web server configurations. We need to make sure that we add the X-Content-Type-Options header with the value nosniff to our configurations. This will stop the mime-sniffing. We will add these heading configurations in the httpd.conf file for the apache configurations.

Strict Transport Security Header Not Set (3): To mitigate this vulnerability, we will need to add the Strict Transport Security header into our web server configurations. This will make the browser only connect to the web servers through HTTPS. So enable HSTS in the web server configurations to make sure that the browsers will only connect using HTTPS. We set the parameters max-age and includeSubDomains directives to enforce HSTS.

Conclusion

In conclusion, the vulnerability assessment conducted on Agricore's infrastructure has provided valuable insights into the security posture of the network. The findings were discovered using both Nessus and NMAP scanning tools to identify vulnerabilities. The vulnerabilities encompass various aspects such as related to SSL, DNS, and other various services involved in the domain. The identified vulnerabilities pose potential risks to the confidentiality, integrity, and availability of Agricore's systems. The explained mitigation strategies provided offer concrete steps and configuration adjustments that could be implemented to enhance the overall security posture of Agricore's infrastructure.